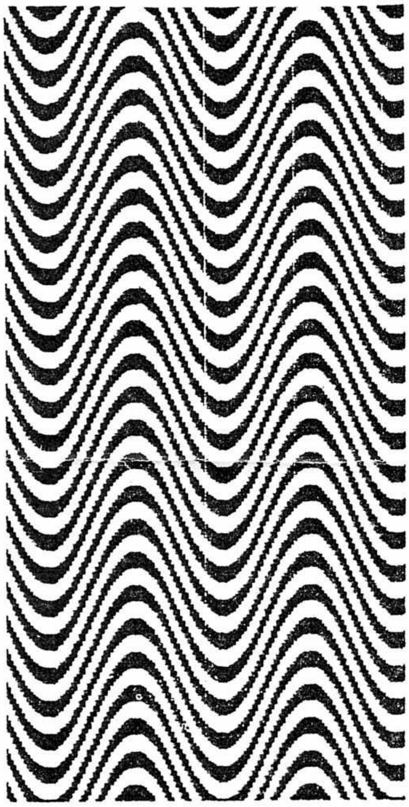
DATA CHIP

Newsletter of the Rochester, NY
Compucolor Users Group

Issue: 31

Apr-Jul '82

Welcome back, Data Chip! After a long absence somewhere, DC is now back in action! Let's not say just where, but with the editor busy and lazy, it's taken a while to get this issue together.



One of the articles is from a sister publication in England. The article about disk setup comes from the CompUKolour Newsletter of the Compucolor Users Group of 19, Harwood Avenue, Bromley, Kent. BR1 3DX. It should be of interest to many of our readers who have experienced "tuning" and compatibility problems with their drives. Also from the same group, a Special Menu program. They are going great guns over there!

Art Buchwald writes about my wife's problem, and Dale Dewey talks about the v6-78 ROM. (If you have fixes for the ROM, submit them to Datachip [or Dale directly], and we'll compile a list of problems/fixes for publication.) Myron Steffy submits an article about Tom Devlin's RAM board. By the way, if you haven't seen it, check the ad in this issue or the article in the Apr/May Colorcue. Tom is finally adding hardware the way we'd like to see it done. Professionally and inexpensively. So are several other individuals and companies (most of them are overseas....). The additions really increase the utility of the good old CCII.

Let's welcome a new Librarian. Doug Van Putte is a frequent contributor to Compucolor publications (see his article on 3-D graphics in the Feb/Mar Colorcue). He's had his machine about as long as anyone, and brings enthusiasm and a new cover page.

By the way, to keep Data Chip coming out, your articles are sorely needed. I have less time for writing now than I did, and can't fill the gaps as well. So, share your experiences, your knowledge. Write!

Membership
Gene Bailey
28 Dogwood Glen
14625

Librarian
Doug Van Putte
18 Cross Bow Dr.
14624

Editor
Ben Barlow
161 Brookside
14618

(All Rochester)

A 'JUMPER' FOR ADDITIONAL ESCAPES

by Myron T. Steffy, Sun City, Arizona

Recently I invested in Tom Devlin's 8K RAM board which installs in the space originally provided for an 8K PROM at 4000H. I say 'invested' advisedly as this addition to my 6.78 Compucolor has been a large plus. It has been well described elsewhere so I won't repeat it here. I have long wanted the ability to have a Screen Editor and an Assembler existing simultaneously in RAM so that I could switch back and forth without reloading either.

This was accomplished by moving the Screen Editor to 4000H but keeping its buffers high enough in memory (9D00H) to clear the Assembler which orginates at 8200H. In essence, both use the same buffer space alternately while not tramping on each other's program listing.

The 6.78 version of Basic had Escapes at only four spots: 8200H, 9000H, A000H and 'User'. The latter is of course programable by the operator. This arrangement requires that you use the programable spot to access 4000H. This is not always desirable as you frequently need it for other programs, possibly including the one you are assembling.

Gordon Rusch (COMTRONICS) has a set of two replacement chips for UA-6 and UA-7 on the logic board, which among other things will give you all of the Escapes provided in the 8.79 Basic plus a few extras. If you don't have them, you will need a way to extend your present group of Escapes. The source code following will do this for you.

If you examine the source code, you will see that it includes all of the 'Escapes' provided in the 8.79 system plus a couple of extras peculiar to my needs. Of course you can extend the list making certain however that the letter or figure you select is not used for other purposes. Consult the escape table in your operating manual. This listing is set up to originate high in memory at FD00H. I have purposely left extra space above it to accomodate the'stacks' of other programs which are sometimes left to be automatically placed at FFFFH. Escape 'D' will return you to FCS control.

For my own use, I attached 'Jumper' to my version of the Screen Editor which loads at 4000H. If you do this, be sure to 'Save' the conglomerate with a 'Start' address at the beginning of 'Jumper', in my instance, 5300H. Otherwise you will not be able to access 'Jumper' with 'ESC User', since its start address must be installed at 81C0H. This is done with 'SHLD 81C0H' early in the 'Jumper' startup.

It should also be mentioned here that many programs with entry points other than 8200H, 9000H or A000H make use of 'ESC ^' for re-entry. If this be true, you will have to find this instruction and disable it. Otherwise once you have 'escaped' to another routine, you will not be able to return to 'Jumper'. The easy way to do this is to use 'DBUG' or an equivalent monitor and search for a 'SHLD 81C0H'. Put three '0' or NOP's in this location to knock it out.

A few words of explanation and we'll leave you to 'hunting and pecking'. After the usual start-up mechanics, 'Query' looks for an 'ESC'. If not forthcoming, it jumps to an ASCII decoding mechanism. First a check is made for hex characters at 'LET' and 'DIG'. They are sorted out and converted to alphanumericals by subtracting the appropriate amount. The first character will be the most significant half of the high byte of the jump address. To give it its proper value, it must be shifted left one place just as you do in decimal arithmetic when you multiply by ten. Since we're dealing with hexidecimal numbers, it must be multiplied by 16. In binary you can do this by rotating left four times, which we have done here. It could also have been done by using the instruction 'Add A' four times, same result.

Then we hang around for the next character of the address to be entered. It is converted in the same fashion but this time is simply added to our first value. This gives us the high byte of the new address which then replaces the high byte of 'Exit'. Likewise the last two characters of the address are processed and installed as the low byte at 'NEWADR+1'. The program really never jumps to 'Exit' at all from 'NEWADR'. It could have just as well been 'XXXX' because it is either replaced by a 'new address' or the program goes to an error routine. Kind of sneaky, isn't it? Have fun!

```
;       'JUMPER' ROUTINE FOR ADDITIONAL ESCAPE LOCATIONS
;       by Myron T. Steffy,  Sun City, Arizona 85351 6/21/82
;=============================================================

        READY   EQU     81FFH
        ESC1    EQU     053AH
        OSTR    EQU     33F4H
        KEYTST  EQU     0024H

        ORG     0FD00H

START:  LXI     H,00H
        DAD     SP
        SHLD    FCSSP
        LXI     SP,STACK
        MVI     A,0C3H
        STA     81BFH
        LXI     H,START
        SHLD    81C0H           ;ESCAPE ^ (USER)
        LXI     H,START-1
        SHLD    80ACH           ;PROTECT THIS PROGRAM

QUERY:  LXI     H,MSG01
        CALL    OSTR
        CALL    GETANS
        CPI     27      ;ESCAPE ?
        JNZ     QUEST
        CALL    GETANS
        CPI     'T'     ;ESC T  8200H
        JZ      8200H
        CPI     'I'     ;ESC I  9000H
        JZ      9000H
        CPI     'S'     ;ESC S  A000H
        JZ      0A000H
        CPI     'P'     ;ESC P  4000H
        JZ      4000H
        CPI     'Z'     ;ESC Z  4800H
        JZ      4800H
        CPI     '\'     ;ESC \  5000H
        JZ      5000H
        CPI     ']'     ;ESC ]  5800H
        JZ      5800H
        CPI     'U'     ;ESC U  USER, EB00H (DBUG32)
        JZ      0EB00H
        CPI     'V'     ;ESC V  E000H
        JZ      0E000H
        CPI     '['     ;ESC [  F500H
        JZ      0F500H
        CPI     'D'     ;ESC D  RETURN TO FCS
        JMP     EXIT

QUEST:  CALL    GTD01
        STA     NEWADR+2        ;LOW BYTE OF JUMP
        CALL    GTDEST
        STA     NEWADR+1        ;HIGH BYTE
NEWADR: JMP     EXIT
```

```
GTDEST: CALL    GETANS
GTD01:  CALL    LET     ;FIRST CHARACTER OF BYTE
        RLC
        RLC
        RLC
        RLC             ;MULTIPLY BY 16
        MOV     B,A     ;STORE IT TEMPORARILY
        CALL    GETANS
        CALL    LET     ;SECOND CHARACTER OF BYTE
        ADD     B
        RET

LET:    CPI     'F'+1   ;CHECK FOR HEX CHARACTER
        JNC     DIG
        CPI     'A'     ;LOW END OF HEX LETTERS
        CMC
        JC      FIXLET
DIG:    CPI     '9'+1   ;CHECK FOR NUMERAL
        JNC     ERMSG
        CPI     '0'     ;LOW END OF NUMERALS
        CMC
        JC      FIXDIG
        JMP     ERMSG   ;IF NONE OF THESE, ERROR!

FIXLET: SUI     37H     ;CONVERT ASCII TO ALPHABETIC
        RET

FIXDIG: SUI     30H     ;CONVERT ASCII TO DECIMAL
        RET

GETANS: MVI     A,32H   ;WAIT FOR ONE CHARACTER
        STA     READY
GETCHA: CALL    KEYTST
        JNZ     GETCHA
        RET

ERMSG:  LXI     H,MSG02 ;PRINT ERROR MESSAGE
        CALL    OSTR
        JMP     QUERY   ;START OVER

EXIT:   LXI     H,MSG03
        CALL    OSTR
        LHLD    FCSSP
        SPHL
        MVI     A,44H
        JMP     ESC1

MSG01:  DB 15,13,10,10,19,9,'ENTER ',21,'ESC ',18,'plus '
        DB 'vector character ',17,9,'OR'
        DB 15,13,10,10,19,9,'ENTER JUMP ADDRESS IN HEX: '
        DB 18,239
MSG02:  DB 17,13,10,10,9,'ERROR, NOT A HEX DIGIT.',18,239
MSG03:  DB 11,28,28,11,239

FCSSP:  DW      01H
        DS      40H
STACK:

        END     START
```

BUCHWALD

Will this marriage fail? Program the variables and see

By ART BUCHWALD

For every home computer sold in America, there is a computer widow somewhere.

I dropped over to see the Bengals the other night. Mrs. Bengal offered me a drink.

"Where's Walter?" I asked Adele.

"Where he always is these nights. He's in the library talking to his home computer."

"He talks to a computer?"

"All the time. It's taken the place of television, conversation and foreplay," she said bitterly.

"I didn't know Walter was into computers."

"That's all he's into. As soon as he finishes dinner, he leaves the table and says, 'Well, I've got to go in and program a new household fiscal budget for 1983.' "

"At least he's working to save you money."

"He says he's working on a new budget, but I walked in last night and he was playing 'Star Wars.' He told me he was just checking out his floppy disk drive. I've never felt so alone in my life. At least when he watched football I could sit next to him. But now that he has a home computer he says he has to be alone with his software."

"You poor kid. Maybe he'll tire of it."

> **"Have you ever thought about getting your own home computer and plugging into his? Perhaps you could talk that way."**

"No way. He reads computer magazines the way he used to read Playboy. His idea of a centerfold now is a 64-K Ram Micro-Computer that will expand to 128 bytes and produce a six-color high graphic screen resolution.

"Has he told you this?"

"No, but he talks in his sleep."

"Well, at least he's not dreaming about another woman," I said.

"I could compete with another woman," Adele said, "But I can't compete with a computer. We have no communication any more. The only language he uses is BASIC, COBOL and FORTRAN. I'm at my wits' end."

"You're not thinking of leaving him?"

"I threatened to last week and he said to hold off until he could program all the variables, and come up with a modified alternative."

"Have you ever thought about getting your own home computer and plugging into his? Perhaps you could talk that way."

"I'm not interested in interfacing with him through a terminal. After all, we're in the same house."

"Maybe I should talk to him," I suggested.

"You can try, but I doubt if it will do any good."

"I went into the library and found Walter hunched over his keyboard. "Hi, Walter. Am I disturbing you?"

"No," he said, squinting at me. "I was only justifying my margins."

"How's life?" I asked.

"Fine. I was having a problem with my cursor for a while, but I straightened it out by adding a protocol."

"You have to be careful of cursors," I said. "What news of Adele?"

"Wait a minute," he said, "I'll find out."

He put in a disk, pushed a code key, and typed on the screen ADELE. Then he hit his RETURN button.

"Here it is," he said. She's either in the kitchen, the bath, her bedroom or went to a baseball game."

"A baseball game?"

Walter looked worried. "That doesn't sound right. But it's no problem. All I have to do is hit this DELETE button."

"Adele thinks she's losing you to a floppy disk retrieval system," I told him.

"That's ridiculous," Walter said. "All I'm trying to do is store and index data that will be able to forecast how we can enjoy the September years of our life."

"We've been friends for years, so I'm going to ask you a very personal question, Walter. How much do you love Adele?"

Walter, without saying a word, inserted a disk, and started hitting the keyboard.

"What are you doing?" I asked.

"I'm counting the ways. It's much faster to do it on a computer."

The Black Arts

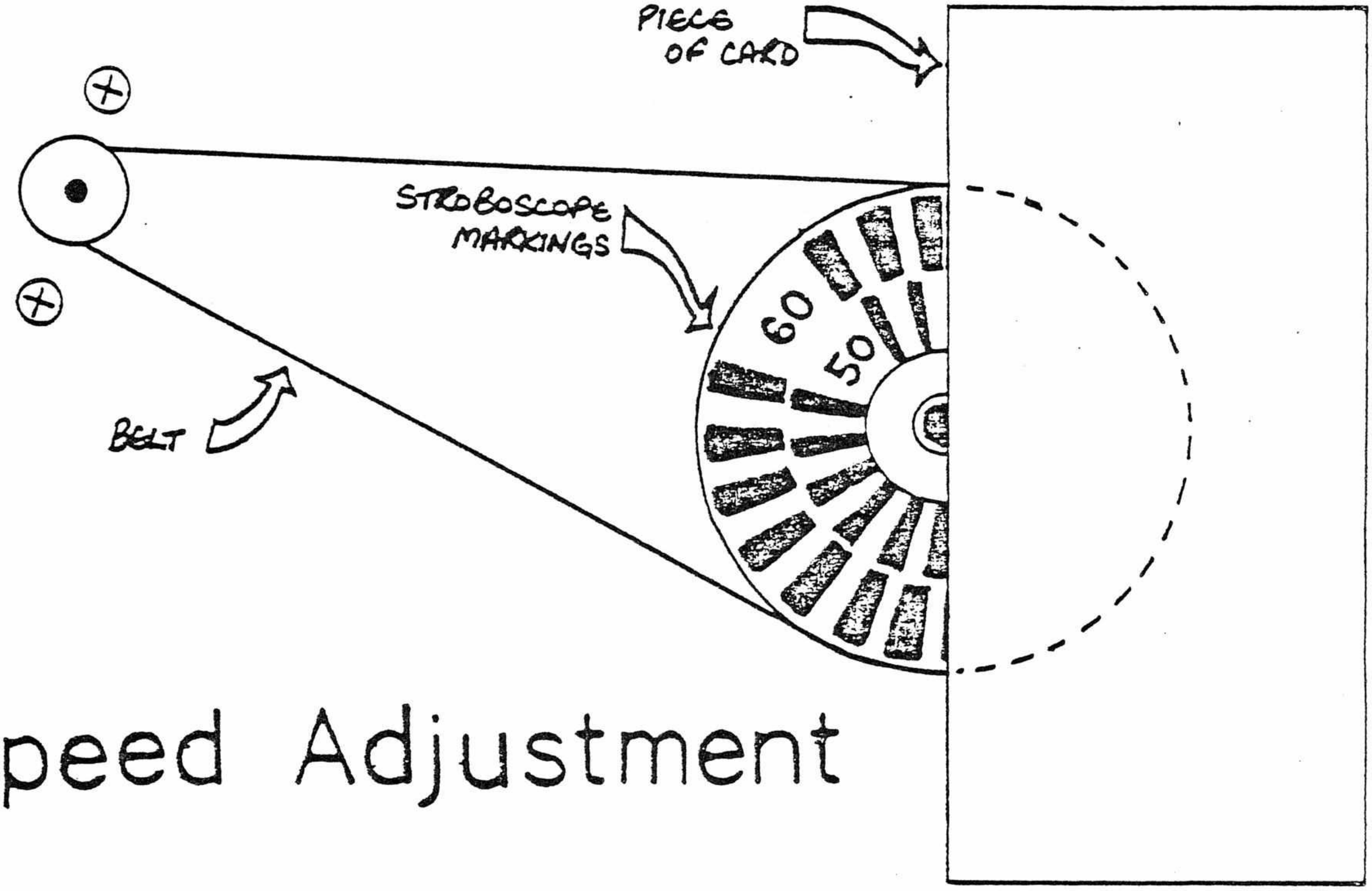
AN OCCASIONAL SERIES FROM THE OTHER SIDE

by Adrian Donkin

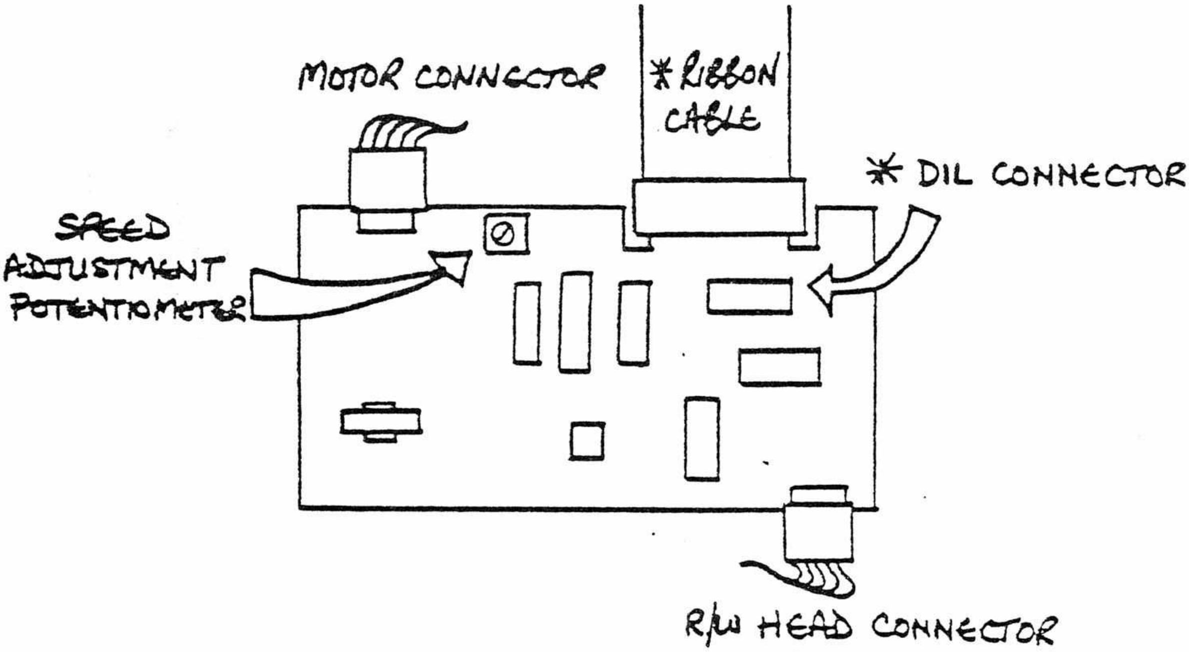
This first hardware article deals with a simple method of disc drive alignment that needs no special tools or specialist electronic equipment. After you have followed these adjustments you will have set the disc drive on your machine to a common standard which will ensure success when exchanging software with other Compucolor users.
I had no problems reading discs made on other machines but when I sent one of my discs to Bill Donkin, he found it almost impossible to read. He suggested that the head alignment was on the outer boundary of the manufactures tolerance and need adjusing. I decided to align my drive and did so using an I.S.C. disc (not a special disc - any one will do, but everyone has the 'Sampler' disc which is a good reasonable reference). I've had no further problems. If you complete the adjustments and find that you read errors on a large number of previously recorded discs, it may be advisable to return your drive to it's previous state of adjustment until you can beg, borrow or steal a second (correctly adjusted) drive on which you can copy your precious discs. If you only have a few errors you might get away with altering the speed before reading the offending disc and then re-setting the speed before writing back to the disc - but it will be hit or miss. My thanks to Stewart Heath of Beorn Designs for the 'sticker idea'.

SPEED ALIGNMENT

Lay the disc drive so that the flywheel with it's stroboscopic markings can be observed. Stand a piece of card in front of the flywheel as shown in the accompanying diagram. This will provide a stationary reference point making it much easier to see when the stroboscopic display is 'stationary' *. Without a disc in the drive unit hit the 'AUTO' key which will set the flywheel into continuous motion. Make a rough assessment of your drive's current speed. (Is the display stationary? Is it moving to the left or to the right? Is it moving quickly or slowly?). Make a note of this assessment, it will help you to return your drive to it's previous condition if after alignment you have difficulty in reading previously recorded discs. Adjust the speed potentiometer (on the disc drive printed curcuit board - see diagram) to halt the stroboscopic display. Check that the display is still stationary after removing your screwdriver as this often disturbs the setting of the speed pot.
* Remember that the stroboscopic display relies on an incandescent lamp which dims sligfhtly twice in every mains cycle. Fluorescent lamps do not do this and are not suitable for this adjustment.



Speed Adjustment



* D.I.L. CONNECTOR USED ONLY ON INTEGRALLY MOUNTED DRIVES, RIBBON CONNECTOR USED ON EXTERNAL DRIVES.

AN OCCASIONAL SERIES FROM THE OTHER SIDE

(contd)

HEAD ALIGNMENT

Make up two stickers of suitable size for mounting on the chassis and stepper motor (see accompanying diagram). Mark the centre of each sticker and apply them to the chassis and stepper motor taking care to line up the two marks.

1. Slacken off the two motor securing bolts (marked 'A' on the diagram) with a suitable Allen key.
2. Insert an I.S.C. produced disc into the drive.
3. Rotate the stepper motor slightly in a clockwise direction and attempt to by pressing Auto to read Menu. Continue turning the motor and reading the disc until a read error occurs. Now turn back the motor very slightly (in a counterclockwise direction) so that an error - free read is just possible and mark the new position of the stepper motor on the chassis sticker (lining up with the mark on the motor sticker). Repeat the whole procedure turning the stepper motor in a counter-clockwise direction and mark the final position on the chassis sticker.

There will be three marks on the chassis sticker. The central mark represents the position of the read/write head prior to adjustment. The two outer marks represent the extreme limits at which the head will still read correctly recorded disc. The optimum position for the head will therefore be midway between the two outer limits. To complete the alignment procedure adjust the stepper motor so that the reference mark on the stepper motor sticker is located midway between the two outside marks on the chassis sticker. Re-tighten the two motor securing bolts (marked 'A' on the diagram).

APPENDIX A - DISASSEMBLY

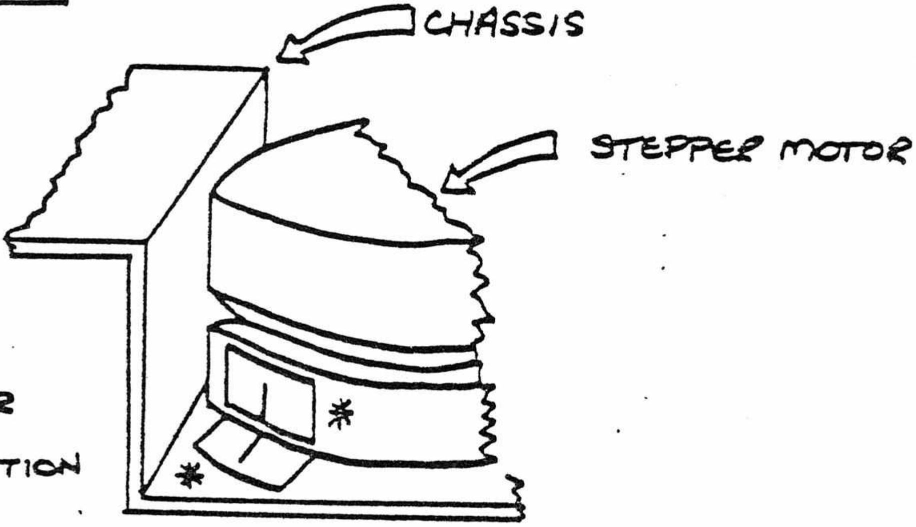
A.1 Internally mounted disc drives.

Remove all connections from the rear of the main cabinet. Place the cabinet 'screen down' on a soft surface. Remove the slotted self-tapping screw marked 'B' on the accompanying diagram. Return the cabinet to it's normal position and remove the three 'Phillips' type bolts (marked 'A' on the diagram. Carefully detach the rear portion of the cabinet placing it so that access to the internals of the machine is possible while the wiring that connects the rear portion to the rest of the cabinet is not strained. Carefully remove the dual-in-line (d.i.l) connector (the gold plated pins are easily bent or broken) from the socket situated behind the keyboard edge connector on the main logic board (make a note of the orientation of the connector for your own reference when you reassemble the machine). Remove the slotted bolt marked 'C' on the diagram. The complete disc drive assembly can now be removed from the front of the cabinet. The drive rests on the bottom edge of its front panel and levers out of the cabinet top edge first. A light push from the rear may assist in its removal. Reconnect the keyboard to the edge connector on the logic board. Finally reconnect the a.c. power lead to the rear portion of the cabinet.

A.2 External disc drives

Unplug the combined keyboard/disc drive connector from the rear of the cabinet. Invert the disc drive cabinet and remove the two visible 'Phillips' type bolts (marked 'A' on the accompanying diagram). Slide the cover off toward the rear of the drive. Remove the two countersunk bolts (marked 'B' on the diagram). Reconnect the keyboard/ disc drive connector to the rear of the cabinet.

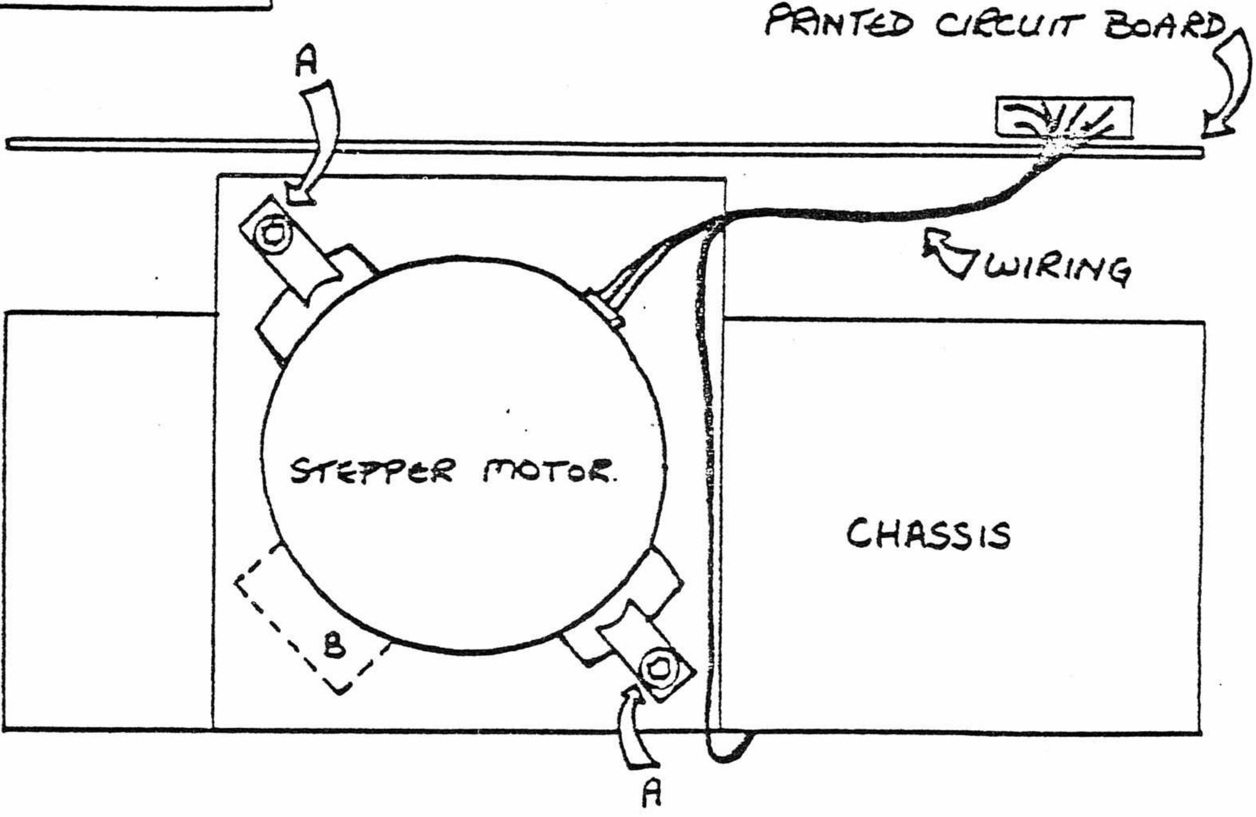
STEPPER MOTOR
- SIDE VIEW



* STICKERS ON
CHASSIS AND MOTOR
MARKING INITIAL POSITION

Head Alignment

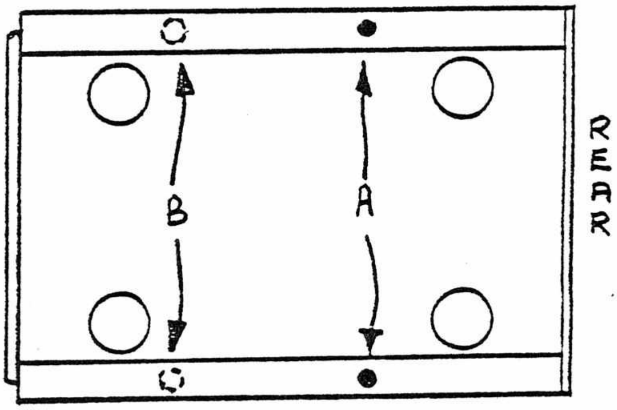
DISC DRIVE
-REAR VIEW



A - ALLEN BOLTS SECURING MOTOR
B - POSITION FOR STICKER ON CHASSIS

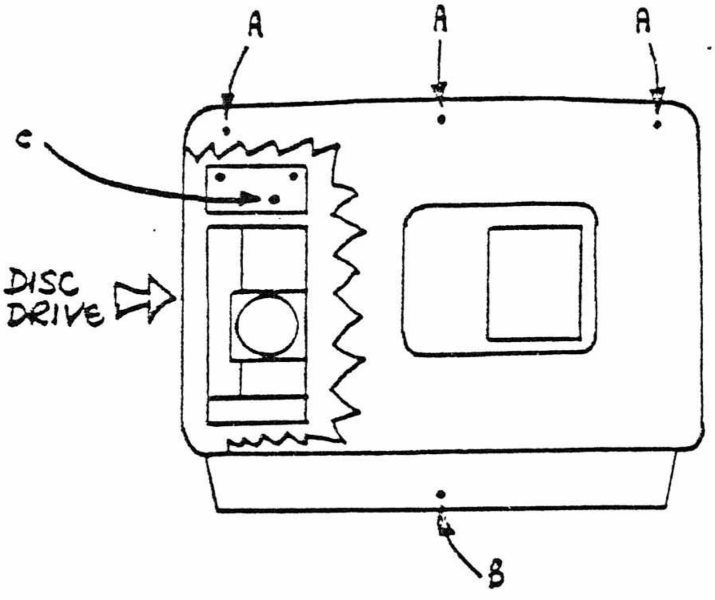
EXTERNAL DISC DRIVE
- BOTTOM VIEW

A - COVER RETAINING BOLTS - PHILLIPS TYPE
B - BASEPLATE RETAINING BOLTS - PHILLIPS C/SUNK



Disassembly

CC II CABINET
-REAR VIEW



CABINET FIXINGS

A - PHILLIPS BOLT
B - SLOTTED SCREW

DRIVE FIXING

C - SLOTTED BOLT

```
100 REM   SPECIAL MENU FOR DIRECTORY LOADING
110 REM   BY LARRY AND CARY PETERSON
120 REM   SITTINGBOURNE KENT    (20 MAR 1982)
130 PLOT 18,12,15 : T=29312
140 REM  LOAD THE DIRECTORY
150 PLOT 27,4 : PRINT "DIR" : PLOT 27,27,5
160 REM  SET UP DISPLAY
170 FOR X=1 TO 20 : GOSUB 290 : T=T+128 : NEXT X
180 T=29440 : GOSUB 330 : GOSUB 370 : GOSUB 400
190 REM  GET KEYBOARD INPUT
200 PLOT 3,65,0
210 POKE 33278,0 : PLOT 16
220 IF PEEK(33278)=0 THEN 220
230 A=PEEK(33278)
240 PLOT 8,18
250 IF A=85 GOTO 460
260 IF A=68 GOTO 500
270 IF A=76 GOTO 540
275 IF A=35 GOTO 700
280 GOTO 190
290 REM  ERASE ARROW
300 POKE T+1,1 : POKE T+3,1 : POKE T+5,1
310 POKE T,32 : POKE T+2,32 : POKE T+4,32
320 RETURN
330 REM  DISPLAY ARROW
340 POKE T,103 : POKE T+1,103 : POKE T+2,103 : POKE T+3,103
350 POKE T+4,62 : POKE T+5,103
360 RETURN
370 REM  ERASE DIRECTORY HEADINGS
380 PLOT 8,10,11,10,11,10,11,10,11
390 PLOT 8 : RETURN
400 REM  PRINT MENU HEADINGS
410 PLOT 8 : PRINT "                 UTILITY MENU"
420 PRINT "      U = MOVE UP"
430 PRINT "      D = MOVE DOWN       £ = DELETE FILE"
440 PRINT "      L = LOAD SELECTION"
445 PLOT 11 : PRINT  : PLOT 11
450 RETURN
460 REM  MOVE ARROW UP
470 GOSUB 290
480 T=T-128 : PLOT 17 : GOSUB 330
490 GOTO 190
500 REM  MOVE ARROW DOWN
510 GOSUB 290
520 T=T+128 : GOSUB 330
530 GOTO 190
540 REM  LOAD SELECTED PROGRAM
550 FOR X=8 TO 32 STEP 2
560 A=PEEK(T+X)
570 A$=CHR$(A)
580 IF A$=" " THEN 620
590 B$=B$+A$
600 IF F=1 THEN F=0 : IF A$="P" THEN 650
610 IF A$="." THEN F=1
620 NEXT X
630 PLOT 6,2,12 : LOAD B$ : RUN
640 END
650 A=LEN(B$) : A$=MID$(B$,1,A-2)
660 C$="RUN "
670 B$=C$+A$
680 PLOT 6,2,12,27,4 : PRINT B$
690 END
700 REM  DELETE A PROGRAM
710 FOR X=8 TO 32 STEP 2
720 A=PEEK(T+X)
730 A$=CHR$(A) : IF A$=" " GOTO 750
740 B$=B$+A$
750 NEXT X
760 C$="DEL " : A$=C$+B$
765 PLOT 3,0,28 : PRINT  A$ : INPUT "OK (Y OR N)? ";V$ : IF  V$=
"Y"  GOTO 770
767 GOTO 780
770 PLOT 6,2,12,27,4 : PRINT  A$
775 PLOT 27,27
780 T=29312 : A$="" : B$="" : C$="" : GOTO 170
```

Dale E. Dewey

D2 Engineering
7284 High View Trail
Victor, New York
14564   U.S.A.

April 4, 1982

Ben Barlow, Editor
Colorcue
161 Brookside Drive
Rochester, New York
14618

Dear Ben:

It has been a while since I talked with you, so I decided to write and let you know what has been going on. First of all, you and Dave are doing a fine job on the Colorcue. It seems that more and more people are getting involved with the poor, outcast COMPUCOLOR II. Now for some updates.

I have finished my first ASSEMBLER PROGRAMMING seminar with good results. Everyone learned a lot about our machine as well as programming. Wouldn't you know, about two weeks into the class my machine went "blue" smoke but we worked around the problems.

I had the 8" disk drives running about the end of January, but decided to drop the whole project. After looking at the amount of hardware I had to collect, I decided none of the users would put up with the cost. I am, however, still working on the STD BUS interface for the CC-II. It is coming along well and I should have a working model by the end of this month.

While my machine was in Georgia being fixed, I did some work on the FCS. After fixing some bugs, I called around and everone said "Sure, so-and-so found that too". Funny, I have never seen any published accounts of bugs in FCS so here are my offerings.

First, COPY does not work. The fix is simple. The V6.78 ROM has a bad location in it. The content of location 2B2BH has a value of 02H in it. This location should contain 22H. Change it and the copy function will work just fine. You will be able to copy one file to another file with a new name on the same disk. Many other combinations are also possible.

Secondly, the famous BELL only chirps. This fix is also simple. The V6.78 ROM has a bad jump address in it. The content of location 3ACFH has a value of 0D8H in it. This location should contain 0DBH. Change it and the BELL will give a nice long 150 millisecond burst of sound.

Another fix I made is really an upgrade. I did not like having to type RUN to get a .PRG program to execute. It always seemed like extra input. The attached listing shows how to modify FCS so it will look for a file on the disk rather than generating the EIVC error code. This works just fine and will generate a file not found (EFNF) error if the file can't be found. The only short comming is that your disk programs can't have the same first three letters as a FCS command. For example, a program WRITER must be run by using the FCS command RUN WRITER since WRI is a command. You could also type 0:WRITER and get the program to run.

I have not done any work on other versions of FCS. I don't have access to a machine other than my own. I have put these changes into TI TMS-2532 ROM chips and all work very well. Is there any interest out there in these changes? I can put the changes into ROMs for anyone who is interested. Also, what about the other versions of FCS? How about a BELL kit? I would be willing to work with anyone who has an interest in improving his little maching so that at least a control G will do something useful. Who has a list of FCS bugs? Let's not hide this stuff. We can all make use of information that effects the general operation of the machine.

Reguards,

Dale Dewey

Dale Dewey

```
  LOC  OBJ         LINE        SOURCE STATEMENT

  25B7              852          ORG     FCSORG
                    853
  25B7 211426       854 CRTMO:   LXI     H,CRTMSG          ;CRT MODE MESSAGE
  25BA CDF433       855          CALL    OSTR
  25BD 2A3600       856          LHLD    ACRTSP            ;SET THE STACK
  25C0 F9           857          SPHL
  25C1 213800       858          LXI     H,BEGEX           ;GOTO EXECUTIVE LOOP
  25C4 E5           859          PUSH    H
  25C5 C9           860          RET
                    861
  25C6              862          DS      4                 ;EXTRA ROOM
                    863
  25CA 13           864 F3:      INX     D                 ;SKIP TO . . .
  25CB 05           865          DCR     B                 ; . . . NEXT . . .
  25CC F2CA25       866          JP      F3                ; . . . COMMAND
  25CF E1           867          POP     H                 ;GET COMMAND LINE PTR
  25D0 1A           868          LDAX    D                 ;GET BYTE
  25D1 A7           869          ANA     A                 ;END OF COMMAND TABLE?
  25D2 C2F325       870          JNZ     F1                ;NO:  TRY NEXT COMMAND
  25D5 EB           871          XCHG                      ;YES: SAVE COMMAND LINE PTR
  25D6 015629       872          LXI     B,RUNOO           ;ASSUME 'RUN'
  25D9 C5           873          PUSH    B
  25DA 21F780       874          LXI     H,FPB             ;POINT TO SYSTEM FPB
  25DD 22F380       875          SHLD    FPBP
  25E0 EB           876          XCHG                      ;GET COMMAND LINE PTR
  25E1 C36034       877          JMP     SPNOR             ;GO TO 'RUNOO'
                    878
  25E4 C5           879 FCSEX:   PUSH    B                 ;SAVE STATUS CODE
  25E5 CDA526       880          CALL    RESET             ;RESET DEVICES
  25E8 C1           881          POP     B                 ;RESTORE STATUS CODE
  25E9 78           882          MOV     A,B               ;GET STATUS CODE
  25EA A7           883          ANA     A                 ;TEST IT
  25EB C9           884          RET                       ;RETURN TO CALLER
                    885
  25EC 11E425       886 FCS:     LXI     D,FCSEX           ;POINT TO EXIT ROUTINE
  25EF D5           887          PUSH    D
  25F0 117A25       888          LXI     D,CMTAB           ;POINT TO COMMAND TABLE
  25F3 E5           889 F1:      PUSH    H                 ;SAVE COMMAND LINE PNTR
  25F4 0603         890          MVI     B,3               ;SET COUNTER
  25F6 1A           891 F2:      LDAX    D                 ;GET NEXT COMMAND CHARACTER
  25F7 13           892          INX     D
  25F8 BE           893          CMP     M                 ;MATCH?
  25F9 23           894          INX     H
  25FA C2CA25       895          JNZ     F3                ;NO
  25FD 05           896          DCR     B                 ;YES: ENOUGH?
  25FE C2F625       897          JNZ     F2                ;NO: CHECK NEXT CHARACTER
  2601 C1           898          POP     B                 ;YES: CLEAR STACK
  2602 EB           899          XCHG                      ;GET COMMAND TABLE PTR
  2603 4E           900          MOV     C,M               ;GET LO BYTE OF ROUTINE ADDRESS
  2604 23           901          INX     H
  2605 46           902          MOV     B,M               ;GET HI BYTE OF ROUTINE ADDRESS
  2606 C5           903          PUSH    B                 ;PUT ADDRESS ON STACK
  2607 21F780       904          LXI     H,FPB             ;POINT TO SYSTEM FPB
  260A 22F380       905          SHLD    FPBP
  260D EB           906          XCHG                      ;GET COMMAND LINE PTR
  LOC  OBJ           LINE        SOURCE STATEMENT

  260E CD7E34         907           CALL    LTNOR               ;SKIP TRAILING LETTERS
  2611 C36034         908           JMP     SPNOR               ;SKIP SPACES & GO TO ROUTINE
                      909
  2614 06             910 CRTMSG:   DB      6,2,11,29
  2615 02
  2616 0B
  2617 1D
  2618 43525420       911           DB      'CRT MODE'
  261C 4D4F4445
  2620 16             912           DB      22
  2621 20563438       913           DB      ' V482'
  2625 32
  2626 12             914           DB      18,10,11,239
  2627 0A
  2628 0B
  2629 EF
                      915
  262A CDEC25         916 FCSEM:    CALL    FCS                 ;PROCESS COMMAND LINE
  262D 78             917 EMESS:    MOV     A,B                 ;COPY STATUS CODE
  262E A7             918           ANA     A                   ;TEST IT
  262F C8             919           RZ                          ;NO ERRORS
  2630 F5             920           PUSH    PSW                 ;SAVE ERROR CODE & FLAGS
  2631 214826         921           LXI     H,FERS1
  2634 CDF433         922           CALL    OSTR                ;PRINT NESSAGE
  2637 215726         923           LXI     H,FERS2-3
  263A 48             924           MOV     C,B
  263B 0600           925           MVI     B,0
  263D 09             926           DAD     B
  263E 0603           927           MVI     B,3
  2640 CDC034         928           CALL    PSTR                ;PRINT ERROR CODE
  2643 CD8B33         929           CALL    CRLF
  2646 F1             930           POP     PSW                 ;RESTORE ERROR CODE & FLAGS
  2647 C9             931           RET
                      932 $NOLIST
                      934 $LIST EJECT
  LOC  OBJ         LINE        SOURCE STATEMENT

                   1608 ;  COPY A FILE
                   1609
 2B20 22A181       1610 COPOO:  SHLD    XFBLK           ;SAVE STRING PNTR
 2B23 CD6E30       1611         CALL    PNFSP           ;GET SOURCE NAME
 2B26 D8           1612         RC                      ;ERROR
 2B27 CD0C2C       1613         CALL    GETTO           ;GET "TO"
 2B2A C0           1614         RNZ                     ;ERROR
 2B2B 22AB81       1615         SHLD    TMP1            ;SAVE STRING PNTR
 2B2E CD8730       1616         CALL    PCFSP           ;GET DESTINATION NAME
 2B31 D2382B       1617         JNC     COPO1           ;NO ERROR
 2B34 3E15         1618         MVI     A,EMFN
 2B36 B8           1619         CMP     B               ;MISSING FILE NAME?
 2B37 C0           1620         RNZ                     ;NO: ERROR
 2B38 CDE726       1621 COPO1:  CALL    CKEND           ;END OF LINE?
 2B3B C0           1622         RNZ                     ;NO: ERROR
 2B3C 2AA181       1623         LHLD    XFBLK           ;GET SOURCE PNTR
 2B3F CD6E30       1624         CALL    PNFSP           ;GET SOURCE NAME
 2B42 CD772C       1625         CALL    CPYDV           ;COPY DEVICE INFO
 2B45 AF           1626         XRA     A               ;SET OPEN TYPE CODE
 2B46 CD862C       1627         CALL    OPENX           ;OPEN SOURCE FILE
 2B49 D8           1628         RC                      ;ERROR
 2B4A 110381       1629         LXI     D,FSBK          ;SET SOURCE PNTR
 2B4D 21A181       1630         LXI     H,XFBLK         ;SET DESTINATION PNTR
 2B50 060A         1631         MVI     B,FDBK-FSBK     ;SET COUNT
 2B52 CD4434       1632         CALL    MOVHD           ;COPY FILE INFO
 2B55 11F780       1633         LXI     D,FPB           ;POINT TO FPB
 2B58 2AAB81       1634         LHLD    TMP1            ;GET STRING PNTR
 2B5B CD8730       1635         CALL    PCFSP           ;GET DESTINATION NAME
 2B5E 3E01         1636         MVI     A,FNEW          ;SET OPEN TYPE CODE
 2B60 CD862C       1637         CALL    OPENX           ;OPEN DESTINATION FILE
 2B63 D8           1638         RC                      ;ERROR
 2B64 2AA381       1639         LHLD    XFBLK+FSIZ-FSBK ;GET SOURCE FILE SIZE
 2B67 220F81       1640         SHLD    FAUX            ;SET FILE SIZE
 2B6A EB           1641         XCHG
 2B6B 2A0581       1642         LHLD    FSIZ            ;GET "FREE" SIZE
 2B6E CD4D34       1643         CALL    CMPHD           ;WILL IT FIT?
 2B71 0636         1644         MVI     B,EWSZ          ;SET ERROR CODE
 2B73 D8           1645         RC                      ;NO: TOO BIG
 2B74 D5           1646         PUSH    D               ;SAVE FILE SIZE
 2B75 11A581       1647         LXI     D,XFBLK+FLBC-FSBK       ;SET SOURCE PNTR
 2B78 210781       1648         LXI     H,FLBC          ;SET DESTINATION PNTR
 2B7B 0606         1649         MVI     B,FDBK-FLBC     ;SET COUNT
 2B7D CD4434       1650         CALL    MOVHD           ;COPY FILE INFO BACK
 2B80 D1           1651         POP     D               ;GET FILE SIZE
 2B81 CD1A2C       1652         CALL    MCHNK           ;COPY THE FILE
 2B84 0645         1653         MVI     B,ECOP          ;SET ERROR CODE
 2B86 D8           1654         RC                      ;ERROR
 2B87 21F780       1655         LXI     H,FPB           ;POINT TO FPB
 2B8A C35928       1656         JMP     CLX             ;CLOSE FILE & EXIT
                   1657
 2B8D CD003F       1658 ERAS:   CALL    SAVE            ;SAVE REGISTERS
 2B90 C33638       1659         JMP     ERASE           ;ERASE SCREEN AND RETURN
                   1660 $EJECT
 LOC  OBJ          LINE        SOURCE STATEMENT

 3AC3 3E40         4642 BEL:     MVI      A,40H
 3AC5 CDE63A       4643          CALL     TIM4Z
 3AC8 3A1B00       4644 BEL1:    LDA      TPROG
 3ACB 32FD81       4645          STA      MS150
 3ACE C3DB3A       4646          JMP      TIM4X1
                   4647
 3AD1 CDD03F       4648 TIM4X:   CALL     SAVE              ;SAVE ALL REGISTERS
 3AD4 21FD81       4649          LXI      H,MS150
 3AD7 34           4650          INR      M
 3AD8 CAE03A       4651          JZ       TIM4Y
 3ADB 3EC3         4652 TIM4X1:  MVI      A,195
 3ADD D30C         4653          OUT      TIME4
 3ADF C9           4654          RET
```

Add 8K of RAM POWER to your CC-II

(That's a possible total of 40K, Folks!)

Tired of reloading utility Programs?, of going 'OM' ??, of switching disks time after time??? Our add on RAM card puts 8K of RAM in the 4000-5FFFH memory area allowing you to 'hide' your utility programs and free up your user RAM in its entirety.

A number of very powerful utililty programs are available for the RAM card.(Check out the offerings from Quality Software, Helm's Software, Mr. Bill Green and Com-Tronics.) For even more efficient programing, stack two of the RAM cards and swap utilities with the flip of a switch.

Installation is fairly simple (limited soldering required), complete step by step instructions with diagrams are included. (See COLORCUE April/May 82, FORUM vol.2 no.1, CUVIC April 82.)

The RAM card comes complete (including RAM chips), tested,and ready to install for $65.00 plus $2.50 P&H. (V8.79 owners please see the COLORCUE article or write for more information before ordering.) Buy two and we'll include stacking instructions and toss in the switch and wire. (Previous customers just remind us)

Also available, an EPROM version, assembled and tested, for $35.00 ($2.50 P&H). (Does not include EPROMs, forINTEL type 2716 or 2732. Please specify so we can strap the board correctly.)

Shipped via First Class mail in the U.S. and Canada. Air mail to overseas. All checks in U.S. funds drawn on U.S. banks, please. Tom Devlin, 3809 Airport Rd. Waterford, MI. 48095 U.S.A.

COLOR GRAPHICS for Intecolor 3651 and Compucolor II Computers

by David B. Suits

COLOR GRAPHICS for Intecolor 3651 and Compucolor II Computers is intended for those who want to learn the ins and outs of color graphics for the Intecolor 3651 or Compucolor II computer. An introductory knowledge of BASIC programming is assumed.

David Suits has unravelled the mysteries of graphics on the Intecolor 3651 and Compucolor II Computers better, perhaps, than anyone else. Suits guides the reader through the graphics capabilities of these computers with cleverness, style, and touches of genius. His appreciation of these computers infects the reader—and a wonderful infection it is! His writing is clear, understandable, witty, and engaging. The teaching ability of a good teacher shines through.

The depth and subtleties of the graphics capabilities shared with the reader must be experienced to be appreciated. There is so much more to be had if one knows how! The topics covered include:

- Cursor control
- Hiding the cursor
- Page/scroll/vertical modes
- CCI codes
- Color keys
- Double blinking
- Blind cursor addressing
- Double height blind cursor
- Special characters
- Larger characters
- Character & point plotting
- Cursor & plot coordinates
- Removing plotted lines
- Cursive writing
- Bar graphs
- Moving bar graphs
- Vector plotting
- Incremental plotting
- Plotting character strings
- "Plot-English"
- Screen refresh memory
- Peeking & Poking screen RAM
- CRT mode
- CRT plotting
- Special function keys
- Saving & duplicating displays
- Editing displays
- Four-directional scrolling
- Large Chess graphics
- Real-time lunar lander
- Radar simulation
- Animated displays
- Fast dice

COLOR GRAPHICS contains over eighty tutorial programs and supplements the above topics with nine appendicies and an extensive index. It takes up where **BASIC Training for Compucolor Computers** leaves off.

David B. Suits received his Ph.D. in Philosophy in 1977 from the University of Waterloo, Waterloo, Ontario, Canada. He presently teaches Logic and Philosophy at Rochester Institute of Technology, Rochester, New York. He has published articles in both philosophy and computer journals and is the author of a number of Intelligent Systems Corporation's programs, including BOUNCE, MAZEMASTER, LINKO and others. His special interests include artificial intelligence, science fiction, games, and music.

COLOR GRAPHICS for Intecolor 3651 and Compucolor II Computers. D.B. Suits. Joseph J. Charles Publishing Company, 130 Sherwood Dr., P.O. Box 750, Hilton, NY 14468. 8.5" x 11" (21.6 cm x 28 cm), ix + 152 pp. (1981). (Shipping wt. 1 lb. 10 oz. (740 g).).

ORDERING INFORMATION

________	ea.	**COLOR GRAPHICS for Intecolor 3651 and Compucolor II Computers***	@ $15.00	________
________	ea.	Page Mode/Plot Mode Chart, No. 101, pads (50-sheets/pad)** This is a to-scale layout form for graphics and output.	@ $ 4.00	________
________	ea.	**BASIC Training for Compucolor Computers**, J. J. Charles* Also applies to Intecolor 3600 series computers.	@ $14.95	________
			Subtotal:	________
			New York State residents add sales tax:	________
			Total:	________

Please remit payment with order. Satisfaction guaranteed or money refunded if returned within 14 days.
ORDER FROM: Joseph J. Charles Publishing Co., 130 Sherwood Dr., Dept. F, P.O. Box 750, Hilton, NY 14468.

Your Name ______________________________

Your Address ______________ City ______________ State ______________ Zip ______________

Note: *Surface postage included. Extra charge for Air Mail if desired: Mexico, Central America-$3.50; S. America, Western Europe, N. Africa-$6.50; Eastern Europe, Africa, Middle East, Pacific, Australia, New Zealand-$9.00.

**Pads prohibitively expensive to air mail to most countries, also frequently subject to duty.